

II. CLAIM AMENDMENTS

1. (currently amended) A method for routing views in a computer graphical user interface, comprising:

determining a view chain data structure comprising at least three entries, each of said entries comprising an application identifier and a view identifier, a view identified by said view identifier being ~~associated with~~ within an application identified by said application identifier;

passing said view chain data structure to a view router from a first application;

launching a first view based on a first entry in said view chain data structure automatically by said view router;

checking whether ~~unprocessed entries for views not launched~~ remain in said view chain data structure, each said entry for a view not launched specifying a view identifier for a view not yet launched by said view router;

launching a second view based on a second entry in said view chain data structure automatically by said view router when ~~unprocessed entries for views not launched~~ remain in said view chain; and

continuing said first application when no ~~unprocessed entries for views not launched~~ remain in said view chain data structure.

2. (Previously Presented) The method according to claim 1, the method further comprising:

gathering data from said first view and said second view; and passing said data from said view router to said first application or to a subsequent application identified in said view chain data structure.

3. (cancelled).

4. (Previously Presented) The method according to claim 2, wherein said gathered data is organized into a journal list comprising an entry for each view in said view chain data structure.

5. (Previously Presented) The method according to claim 2, wherein said gathered data is organized into a list of type and value pairs.

6. (Previously Presented) The method according to claim 5, wherein said data type and value pair are defined in a markup language format.

7. (Previously Presented) The method according to claim 2, wherein said view router provides a generic interface with generic methods and acts as an adapter for returning data from said first view to said first application or a subsequent application identified in said view chain data structure.

8. (Original) The method according to claim 1, wherein said electronic device has a graphical user interface.

9. (Original) The method according to claim 1, wherein said view comprises user interface elements.

10. (Original) The method according to claim 8, wherein said view is a window opened during said launching step.

11. (Cancelled).

12. (Previously Presented) The method according to claim 1, wherein at least part of said view chain data structure is specified in the memory of said electronic device.

13. (Previously Presented) The method according to claim 12, wherein said view chain data structure is updated based on user actions.

14. (Previously Presented) The method according to claim 1, wherein said view chain data structure is determined based on user actions.

15. (currently amended) A system comprising:

a processor configured to determine a view chain data structure comprising at least three entries, each of said entries comprising an application identifier and a view identifier, a view identified by said view identifier being within an application identified by said application identifier, to pass said view chain data structure to a view router from a first application, to launch a first view based on a first entry in said view chain data structure automatically by said view router, to check

whether entries for views not launched remain in said view chain data structure, each said entry for a view not launched specifying a view identifier for a view not yet launched by said view router, to launch a second view based on a second entry in said view chain data structure automatically by said view router when entries for views not launched remain in said view chain and to continue said first application when no entries for views not launched remain in said view chain data structure; a first application configured to determine a view chain data structure comprising at least three entries, each of said entries comprising an application identifier and a view identifier, a view identified by said view identifier being associated with an application identified by said application identifier, and to pass said view chain data structure to a view router; and

said view router configured to process said view chain data structure, to launch a first view automatically based on a first entry in said view chain data structure, to check whether unprocessed entries remain in said view chain data structure, to launch a second view based on a second entry in said view chain data structure automatically by said view router when unprocessed entries remain in said view chain and to continue said first application when no unprocessed entries remain in said view chain data structure and

a display configured to display said first view and said second view.

16. (currently amended) The system according to claim 15, wherein said view router processor is further configured to gather data from said first view and said second view, and to pass said data to said first application or to a subsequent application identified in said view chain data structure.

17. (Cancelled)

18. (Previously Presented) The system according to claim 16, wherein said gathered data is organized into a journal list comprising an entry for each view in said view chain data structure.

19. (Previously Presented) The system according to claim 16, wherein said gathered data is organized into a list of type and value pairs.

20. (Original) The system according to claim 16, wherein said data type and value pair are in markup language format.

21. (Previously Presented) The system according to claim 15, wherein said view router provides a generic interface with generic methods and acts as an adapter for returning information from said first view to said first application or a subsequent application identified in said view chain data structure.

22. (Original) The system according to claim 15, wherein said system has a graphical user interface.

23. (Original) The system according to claim 15, wherein said view comprises user interface elements.

24. (Original) The system according to claim 22, wherein said view is a window opened during view launching.

25. (Cancelled).

26. (Previously Presented) The system according to claim 15, wherein at least part of said view chain data structure is specified in the memory of an electronic device.

27. (Previously Presented) The system according to claim 26, wherein said view chain data structure is updated based on user actions.

28. (Previously Presented) The system according to claim 15, wherein said view chain data structure is determined based on user actions.

29. (currently amended) An ~~electronic device~~ apparatus comprising:

a processor configured to determine a view chain data structure comprising at least three entries, each of said entries comprising an application identifier and a view identifier, a view identified by said view identifier being within an application identified by said application identifier, to pass said view chain data structure to a view router from a first application, to launch a first view based on a first entry in said view chain data structure automatically by said view router, to check whether entries for views not launched remain in said view chain data structure, each said entry for a view not launched specifying a view identifier for a view not yet launched by said view router, to launch a second view based on a second entry in said view chain data structure automatically by said view router when entries for views not launched remain in said view chain and to continue said first application when no entries for views not launched remain in said view chain data structure; and a first application configured to determine a view chain data structure comprising at least three entries, each of said entries comprising an

~~application identifier and a view identifier, a view identified by said view identifier being associated with an application identified by said application identifier, and to pass said view chain data structure to a view router; and~~

~~said view router configured to process said view chain data structure, to launch a first view automatically based on a first entry in said view chain data structure, to check whether unprocessed entries remain in said view chain data structure, to launch a second view based on a second entry in said view chain data structure automatically by said view router when unprocessed entries remain in said view chain and to continue said first application when no unprocessed entries remain in said view chain data structure~~

a display configured to display said first view and said second view.

30. (currently amended) The ~~electronic device~~ apparatus according to claim 29, wherein said ~~view router processor~~ is further configured to gather data from first view and said second view, and to pass said data to said first application or to a subsequent application identified in said view chain data structure.

31. (Cancelled).

32. (currently amended) The apparatus ~~electronic device~~ according to claim 30, wherein said gathered data is organized into a journal list comprising an entry for each view in said view chain data structure.

33. (currently amended) The apparatus ~~electronic device~~ according to claim 30, wherein said gathered data is organized into a list of type and value pairs.

34. (currently amended) The ~~apparatus electronic device~~ according to claim 30, wherein said data type and value pair are in markup language format.

35. (currently amended) The ~~apparatus electronic device~~ according to claim 29, wherein said view router provides a generic interface with generic methods and acts as an adapter for returning information from said first view to said first application or a subsequent application identified in said view chain data structure.

36. (currently amended) The ~~apparatus electronic device~~ according to claim 29, wherein said ~~apparatus electronic device~~ has a graphical user interface.

37. (currently amended) The ~~apparatus electronic device~~ according to claim 29, wherein said view comprises user interface elements.

38. (currently amended) The ~~apparatus electronic device~~ according to claim 29, wherein said view is a window opened during view launching.

39. (Cancelled).

40. (currently amended) The ~~apparatus electronic device~~ according to claim 29, wherein at least part of said view chain data structure is specified in the memory area of said electronic device.

41. (currently amended) The ~~apparatus electronic device~~ according to claim 29, wherein said view chain data structure is updated based on user actions.

42. (currently amended) The ~~apparatus electronic device~~ according to claim 29, wherein said view chain data structure is determined based on user actions.

43. (currently amended) A computer readable storage medium having a computer program embodied thereon, the computer program comprising code for controlling a processor to execute a method comprising:

determining a view chain data structure comprising at least three entries, each of said entries comprising an application identifier and a view identifier, a view identified by said view identifier being within an application identified by said application identifier;

passing said view chain data structure to a view router from a first application;

launching a first view based on a first entry in said view chain data structure automatically by said view router;

checking whether entries for views not launched remain in said view chain data structure, each said entry for a view not launched specifying a view identifier for a view not yet launched by said view router;

launching a second view based on a second entry in said view chain data structure automatically by said view router when entries for views not launched remain in said view chain; and

~~continuing said first application when no entries for views not launched remain in said view chain data structure determining a view chain data structure comprising at least three entries, each of said entries comprising an application identifier and a view identifier, a view identified by said view identifier being associated with an application identified by said application identifier;~~

~~passing said view chain data structure to a view router from a first application;~~

~~launching a first view based on a first entry in said view chain data structure automatically by said view router;~~

~~checking whether unprocessed entries remain in said view chain data structure;~~

~~launching a second view based on a second entry in said view chain data structure automatically by said view router when unprocessed entries remain in said view chain; and~~

~~continuing said first application when no unprocessed entries remain in said view chain data structure.~~

44. (Cancelled)

45. (Previously Presented) The computer readable storage medium according to claim 43, wherein said computer readable storage medium is a removable memory card.

46. (Previously Presented) The computer readable storage medium according to claim 43, wherein said computer readable storage medium is a magnetic or optical disk.

47. (Previously Presented) The computer readable storage medium according to claim 43, comprising code for controlling a processor to execute a method further comprising:

gathering data from said at least one first view and said second view; and passing said data from said view router to said first application or to a subsequent application identified in said view chain data structure.

48. (Cancelled).

49. (Previously Presented) The computer readable storage medium according to claim 47, wherein said gathered data is organized into a journal list comprising an entry for each view in said view chain data structure.

50. (Previously Presented) The computer readable storage medium according to claim 47, wherein said gathered data is organized into a list of type and value pairs.

51. (Previously Presented) The computer readable storage medium according to claim 50, wherein said data type and value pair are in a markup language format.

52. (Previously Presented) The computer readable storage medium according to claim 47, wherein said view router provides a generic interface with generic methods and acts as an adapter for returning data from said first view to said first application or a subsequent application identified in said view chain data structure.

53. (Previously Presented) The computer readable storage medium according to claim 43, wherein said computer program has a graphical user interface.

54. (Previously Presented) The computer readable storage medium according to claim 43, wherein said view comprises user interface elements.

55. (Previously Presented) The computer readable storage medium according to claim 53, wherein said view is a window opened during said launching step.

56. (Previously Presented) The computer readable storage medium according to claim 43, wherein at least part of said view chain data structure is specified in the memory of an electronic device.

57. (Previously Presented) The computer readable storage medium according to claim 56, wherein said view chain data structure is updated based on user actions.

58. (Previously Presented) The computer readable storage medium according to claim 43, wherein said view chain data structure is determined based on user actions.

59. (Cancelled)

60. (Cancelled)

61. (Previously Presented) The computer readable storage medium according to claim 43, wherein said view router is implemented as a library.

62. (Previously Presented) The computer readable storage medium according to claim 43, wherein said view router is implemented as an own application.

63. (Cancelled)